

HW1

Clarissa Littler

October 6, 2014

Contents

1	HW 1: Regular Languages	1
1.1	Problem 1	1
1.2	Problem 2	1
1.3	Problem 3	2
1.4	Problem 4	2
1.5	Problem 5	2
1.6	Problem 6	2

1 HW 1: Regular Languages

1.1 Problem 1

Give state diagrams of DFAs for the following languages

- $\{w|w \text{ contains the substring } ab \text{ and } ba\}$
- $\{w|w \text{ contains an even number of 0s or exactly three 1s}\}$
- $\{w|w \text{ is a binary multiple of 5}\}$ Note: there's a trick to this one. As a hint there should be a total of *five* states in your DFA and only one accept state. As another hint, think about reading a binary number from left to right and how you calculate the number as an iterative process.

1.2 Problem 2

Prove or disprove the following: let D be a DFA with k -states. If the language $L(D)$ is *finite* then there exists at least one string s of length at-most $k - 1$ such that D does not accept s . Hint: what do we know about a DFA if its language is finite?

1.3 Problem 3

For any string $w = w_1w_2 \dots w_n$ then $w^R = w_nw_{n-1} \dots w_1$ is the reverse of the string. For any language A , let $A^R = \{w^R | w \in A\}$. Prove that if A is regular then so is A^R .

1.4 Problem 4

Give NFAs for the following languages

- $\{w | w \text{ uses an arbitrary number of all but one of } \{a, b, c, d\}\}$ (Clarification: for the purposes of this language strings such as ab , a , ϵ are all **in** the language. Basically, any string that doesn't use all four letters of the alphabet is in the language.)
- $\{w | w \text{ ends with a zero}\}$ but you must use only two states

1.5 Problem 5

(from 1.38 in Sipser) An all-NFA M is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$ that is like an NFA except that the acceptance condition is that *every* possible state the NFA can be in at the end of processing the string must be an accept state. Prove that all-NFAs are equivalent in power to DFAs. Hint: what does the acceptance condition mean in terms of \exists and \forall and if there are *no* possible states you can be in?

1.6 Problem 6

Prove *using the pumping lemma* that the following languages aren't regular. Remember, your proof needs to be a proof by contradiction in the form we did in class.

- $\{0^n 1^m \mid m > n\}$
- $\{s \mid s \text{ has an even number of 0s and fewer 1s than pairs of 0s}\}$